AMENDMENTS TO THE CLAIMS

Please amend the claims as follows.

1. (Currently Amended) A method for statement boundary detection comprising:

obtaining an input stream;

parsing said input stream to determine a natural end of a first statement using a programming language syntax, wherein using said programming language syntax comprises recognizing said natural end of said first statement with and without said first statement having an explicit statement terminator; ~~and~~

dividing said input stream into a series of statements wherein said natural end is used to divide said first statement from a second statement[[.]]; and

compiling said series of statements to create executable code for execution on a processor of a computer system.

2. (Original) The method of claim 1 wherein said parsing comprises:

retrieving a next character from said input stream; and

positioning said natural end of said first statement immediately before said next character, if appending said next character to said first statement is inconsistent with said programming language syntax.

3. (Original) The method of claim 1 wherein said parsing comprises:

determining a context from one or more characters previously retrieved from said input stream; and

positioning said natural end based on said context and said programming language syntax.

4. (Original) The method of claim 1 further comprising:

detecting a statement termination token.

5. (Currently Amended) A statement boundary detector system comprising:

a processor;

main memory; and

instructions stored in main memory for executing on said processor to:

~~an obtainer configured to~~ obtain an input stream;

~~a parser configured to~~ parse said input stream to determine a natural end of a first statement using a programming language syntax, wherein using said programming language syntax comprises recognizing said natural end of said first statement with and without said first statement having an explicit statement terminator; ~~and~~

~~a partitioning unit configured to~~ divide said input stream into a series of statements wherein said natural end is used to divide said first statement from a second statement;[[.]] and

compile said series of statements into executable code.

6. (Currently Amended) The statement boundary detector system of claim 5 ~~wherein said parser~~ further comprise~~s~~ing instructions stored in main memory for executing on said processor to:

~~a retrieval unit configured to~~ retrieve a next character from said input stream; and

~~a positioning unit configured to~~ position said natural end of said first statement immediately before said next character, if appending said next character to said first statement is inconsistent with said programming language syntax.

7. (Currently Amended) The statement boundary detector system of claim 5 ~~wherein said parser~~ further comprise~~s~~ing instructions stored in main memory for executing on said processor to:

~~a determiner configured to~~ determine a context from one or more characters previously retrieved from said input stream; and

~~a positioning unit configured to~~ position said natural end based on said context and said programming language syntax.

8. (Currently Amended) The statement boundary detector system of claim 5 further comprising instructions stored in main memory for executing on said processor to:

~~a detector configured to~~ detect a statement termination token.

9. (Currently Amended) A computer program product comprising:

a computer usable medium having computer readable program code embodied therein configured to detect a statement boundary, said computer program product comprising:

computer readable code configured to cause a computer to obtain an input stream;

computer readable code configured to cause a computer to parse said input stream to determine a natural end of a first statement using a programming language syntax, wherein using said programming language syntax comprises recognizing said natural end of said first statement with and without said first statement having an explicit statement terminator; and

computer readable code configured to cause a computer to divide said input stream into a series of statements wherein said natural end is used to divide said first statement from a second statement.

10. (Original) The computer program product of claim 9 wherein said computer readable code configured to cause a computer to parse comprises:

computer readable code configured to cause a computer to retrieve a next character from said input stream; and

computer readable code configured to cause a computer to position said natural end of said first statement immediately before said next character, if appending said next character to said first statement is inconsistent with said programming language syntax.

11. (Original) The computer program product of claim 9 wherein said computer readable code configured to cause a computer to parse comprises:

computer readable code configured to cause a computer to determine a context from one or more characters previously retrieved from said input stream; and

computer readable code configured to cause a computer to position said natural end based on said context and said programming language syntax.

12. (Original) The computer program product of claim 9 further comprising:

computer readable code configured to cause a computer to detect a statement termination token.